

VU Research Portal

Deliberative Evolution in Multi-Agent Systems

Brazier, F.M.; Jonker, C.M.; Treur, J.; Wijngaards, N.J.E.

published in

International Journal of Software Engineering and Knowledge Engineering
2001

DOI (link to publisher)

[10.1142/S0218194001000670](https://doi.org/10.1142/S0218194001000670)

document version

Publisher's PDF, also known as Version of record

[Link to publication in VU Research Portal](#)

citation for published version (APA)

Brazier, F. M., Jonker, C. M., Treur, J., & Wijngaards, N. J. E. (2001). Deliberative Evolution in Multi-Agent Systems. *International Journal of Software Engineering and Knowledge Engineering*, 11(5), 559-581.
<https://doi.org/10.1142/S0218194001000670>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:

vuresearchportal.ub@vu.nl

Deliberative Evolution in Multi-Agent Systems

F. M. T. Brazier, C.M. Jonker, J. Treur, and N.J.E. Wijnngaards

Vrije Universiteit Amsterdam, Department of Artificial Intelligence
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands.

Email: {frances, jonker, treur, niek}@cs.vu.nl

URL: <http://www.cs.vu.nl/~{frances, jonker, treur, niek}>

Abstract. Evolution of automated systems, in particular evolution of automated agents based on agent deliberation, is the topic of this paper. Evolution is not a merely material process, it requires interaction within and between individuals, their environments and societies of agents. An architecture for an individual agent capable of (1) deliberation about the creation of new agents, and (2) (run-time) creation of a new agent on the basis of this, is presented. The agent architecture is based on an existing generic agent model, and includes explicit formal conceptual representations of both design structures of agents and (behavioural) properties of agents. The process of deliberation is based on an existing generic reasoning model of design. The architecture has been designed using the compositional development method DESIRE, and has been tested in a prototype implementation.

1. Introduction

Evolution in societies of agents is a challenging phenomenon, both from a fundamental perspective and from an applied perspective. Traditionally, when modelling and simulating evolution, evolution is assumed to be a material process, without taking mental abilities into account. Often genetic programming approaches are used and relatively simple agents are considered, which have no deliberative influence on the direction of the process of evolution; e.g., [1,2]. This results in agents that have limited autonomy viewed over several generations: agents are passive with respect to the evolution process. This is not surprising given that from a historical, biological perspective all species have, in fact, always been subject to almost non-autonomous evolution. The influence of mental processes, for example, in searching and deciding on mating partners should, however, be considered. Moreover, the acquisition of knowledge on the relation between DNA structures and properties of individuals, and techniques to manipulate DNA structures opens the perspective of deliberately influencing the direction of evolution, even more. This perspective introduces serious, unresolved ethical dilemmas especially when applied to human society.

Artificial societies of agents make it possible to experiment with deliberative evolution processes without these ethical dilemmas. Moreover, interesting application areas have emerged: applications

related to the Internet, such as Electronic Commerce, within which the evolution of (societies of) agents can be directly influenced.

This paper introduces an approach to evolution in which a role for both mental aspects (mind) and physical aspects (matter) is modelled. An architecture is presented for agents that are able to deliberately create new agents, or even modify or delete existing agents. For example, Internet agents that are capable of deliberately creating new agents to assist them in information gathering, or agents that are capable of creating interface agents tuned to specific users, are agents of this type.

Deliberative evolution is modelled as a reasoning process about design in which the description of the structure of a new agent is (re-)designed to adhere to (new) requirements. These requirements are expressed in terms of structure and properties of the new agent. The agent responsible for the (re-)design process is able to reason about desired (behavioural) *properties* (requirements) of the new agent (in a metaphoric sense comparable to the traditional biological notion of phenotype), about its (design) *structure* (in a metaphoric sense comparable to the notion of genotype) of an agent, and about the relation between structure and properties. It is also able to generate the goal to create an agent with these intended properties. To actually create an agent according to this goal, an intentional action in the material world is performed by the agent.

In this approach, agents (automated human processes) are autonomous, distributed processes. Each agent has its own environment, consisting of other agents and a material world. Agents are able to communicate with each other, can co-operate to jointly perform tasks, interact with the world (observe/act), and perform specific tasks. This relates to the notions of weak agency by Wooldridge and Jennings [3]: autonomy, social ability, reactivity, and pro-activeness. The notion of strong agency is based on the characteristics of mentalistic and intentional notions, which relates to, e.g., BDI-agents (beliefs, desires, intentions) [4,5].

The architecture introduced in this paper has been designed using the compositional development method for multi-agent systems DESIRE [6], and implemented using the DESIRE software environment. The basic notions deliberation, evolution, and mind-matter relationships are discussed in Section 2. Section 3 provides a global view of the approach to deliberative agent evolution. The deliberation model within an agent is described in Section 4. Section 5 describes the design and implementation of the actual creation of an agent. Section 6 discusses the results and future perspectives.

2 Deliberation, Evolution, and Mind-Matter Relationships

This section discusses the notions, ‘deliberation’, ‘evolution’ and ‘mind-matter’ in relation to automated agent evolution.

2.1 Deliberation

Agents or organisms with mental capabilities can influence their interaction with their environment. Simple, hard-wired, stimulus-response patterns do not require extensive internal mental processing. More complex interactions, however, often require a more conscious *deliberation process* in which reasoning takes place and different options are considered. In this paper our notion of deliberation consists of

- generation of different options for possible goals,
- evaluation of each of these generated options for goals,
- selecting one of the options for goals on the basis of these evaluations,
- generation of different options for actions,
- evaluation of each of these options for actions, for example by inner simulation or what-if reasoning,
- choosing one of the options for actions on the basis of these evaluations.

For each of these activities, a deliberation process often needs explicitly represented knowledge:

- knowledge on relevancy of goals for given circumstances,
- evaluation knowledge for possible goals,
- decision knowledge for given goal evaluations,
- knowledge on relevance of actions for a given goal,
- evaluation knowledge for possible actions, such as simulation knowledge,
- decision knowledge for given action evaluations.

This paper does not claim that all mental processes are deliberative. However, the mental processes modelled within the agents described in this paper have been designed to be of a specific deliberative form.

2.2 Evolution

Different notions of evolution exist within different disciplines (such as Biology, Artificial Intelligence, Computer Science, Social Sciences). These notions all define a central role for a population of individuals, and distinguish generation and selection of new generations of individuals. There are, however, considerable differences as well. To make clear how our notion of evolution is positioned in this spectrum, a number of notions of evolution are considered:

- biological evolution,
- cultural evolution,
- inner mental evolution,
- human evolution,
- evolution by standard genetic algorithms,
- evolution in evolutionary software development processes.

Below the following aspects are addressed for each of these notions:

- *How alternative options for new individuals are generated*
- *How selection criteria are determined*
- *How the actual selection of individuals takes place*

Two factors considered are (1) whether there is an active role for individuals of the population (endogenous), or it is some mechanism from outside (exogenous), and (2) whether evolution is a purely physical/biological (or pseudo-physical) process, or it is making use of mental capabilities of individuals.

2.2.1 Biological evolution

Biological evolution is a (distributed) process of change of populations in nature over time by physical processes of birth and death without any centralised control (human evolution is described in the next section) [7,8].

- *How alternatives for new individuals are generated*

Alternatives for new individuals are generated from within the population by active (e.g., mating, cell division) behaviour of individuals (endogenous). In this process both physical aspects (e.g., food and health situation, the encounters) and, to some extent, mental capabilities (e.g., active focused search for a partner, recognition and choice of a partner, mating skills) play a role.

- *How selection criteria are determined.*

Selection criteria stem mainly from the conditions of nature; since nature is, to a large extent, formed by the living populations themselves as well, selection criteria can be viewed as both endogenous and exogenous.

- *How the actual selection of individuals takes place.*

The actual selection takes place during the lifetime of individuals, and is the result of a dynamic process within the population (endogenous). This process is basically physical and not mental.

2.2.2 Human evolution

Biological evolution for the human population is primarily based on the same biological mechanisms as the evolution process of populations in nature [9], but there are some important additions as well, mainly in the role played by mental processes.

- *How alternatives for new individuals are generated.*

In addition to biological aspects mentioned above, birth control plays a role. This means that in the generation of new individuals different factors can be considered, making the mental aspect more dominant.

- *How selection criteria are determined.*

Mental aspects may play an important role in selection. Certain genetically based diseases may be considered to be undesirable, and thus used as a basis for selection. Gender may be another. In the future, due to extended technical possibilities of genetic engineering, the mental aspect will become even more important: which of these technical possibilities is acceptable to be used for which criterion?

- *How the actual selection of individuals takes place*

On the one hand, selection may take place during the lifetime of individuals, although in Western society the biological ‘selective pressure’ has drastically decreased. On the other hand, selection sometimes takes place not during the actual lifetime of an individual, but before birth (i.e., initiated abortion): in relation to explicitly established criteria, as the result of an explicit mental activity.

2.2.3 Cultural evolution

Cultural evolution is the (distributed) evolution process of, for example, development and transmission of:

- imitation behaviour; e.g., [10,11].
- tool manufacturing and use; e.g., [12],
- language; e.g., [13],
- social structures; e.g., [14],
- art, et cetera.

A basic difference with biological evolution is that results are not carried over to next generations by a biological process (i.e., essentially duplicating the genes), but by another form of materialisation (e.g., tools and art products that remain available for hundreds or thousands of years) or social processes (e.g., imitation and education).

- *How alternatives for new individuals are generated.*

New individuals are shaped by education and other means of transfer. This takes place as a primarily mental (and social) process, by individuals from within the population (endogenous), but physical aspects such as, for example, transfer of tools and art products to next generations are also of importance.

- *How selection criteria are determined.*

Selection criteria are mainly determined by mental activity of individuals of the population itself (endogenous): they determine the value of cultural achievements and whether they are carried over to the next generation. Exogenous physical aspects play a role as well: how persistent are the material cultural products, how vulnerable are they to disasters or wars.

- *How the actual selection of individuals takes place.*

The actual selection is mainly the result of mental processes of individuals within the population (endogenous), both before and after birth of new cultural achievements. In addition, natural disasters may play a role in destructing culture during its lifetime (exogenous), basically physical processes.

2.2.4 Evolution by Standard Genetic Algorithms

Evolution on the basis of standard use of genetic algorithms is a simplified model of biological evolution. Individuals do not play an active role in the process. Individuals basically solely serve as passive data storage of a variety of characteristics [15]. Processes are not distributed but globally controlled outside the population.

- *How alternatives for new individuals are generated.*

Alternatives for new individuals are not generated from within the population by active behaviour of individuals. Instead they are generated by an algorithm outside the population that makes use of the characteristics of individuals within the population (exogenous). This process has no relation to any mentality of individuals; it can be (and, in this paper, is) considered to be purely (pseudo-)physical.

- *How selection criteria are determined*

Selection criteria are exogenous. They are one of the parameters under control of the (human) user of the algorithm. No mentality of individuals involved.

- *How the actual selection of individuals take place*

The actual selection is exogenous as well, for similar reasons as mentioned above. This process can be considered to be (pseudo-)physical (i.e., as a form of simulation of a physical process) and not mental.

2.2.5 Evolution in evolutionary software development processes

Evolutionary software development has been put forward as an alternative for methods based on the classical waterfall model, e.g., [16,17]. The idea is that software evolves permanently, and in principle no distinction is made between development and maintenance (which is considered to be part of continued development). The use of a system detects its shortcomings and generates new requirements that are input to continued development of the system. In this case the set of systems (under development by a given developers team) can be considered as a population. In the discussion below it is assumed that software has no active self-maintenance or self-modification facilities built in. Although this is not a very convincing case of evolution, it is included in the overview in order to compare it to the approach to deliberative evolution proposed in this paper.

- *How alternatives for new individuals are generated.*

Alternatives for new individuals are not generated from within the population by active behaviour of individuals. Instead they are generated by the team of developers outside the population that make use of the characteristics of the software available within the population (reuse, exogenous). This process has no relation to any form of mentality or activity of software individuals.

- *How selection criteria are determined.*

Selection criteria are exogenous. They are parameters under control of the team of developers. No mentality of individuals involved.

- *How the actual selection of individuals takes place*

The actual selection is exogenous as well, for similar reasons as above. This process may be considered to be (pseudo-)physical and not mental.

2.2.6 Summary

A summary of the aspects of the phenomena discussed above, can be found in Table 1. In addition, the deliberative evolution approach is included in this table. The table shows that deliberative evolution incorporates an important role for endogenous aspects (the agents of the population may operate in a distributed and autonomous manner, independent of control from elsewhere). Moreover, mental aspects play an important role: agents may deliberate about the various aspects of the evolutionary process.

	<i>generation</i>				<i>selection criteria</i>				<i>selection</i>			
	endo	exo	mental	physical	endo	exo	mental	physical	endo	exo	mental	physical
biological evolution	+	-	+	+	+	+	-	+	+	-	-	+
human evolution	+	-	+	+	+	+	+	+	+	-	+	+
cultural evolution	+	-	+	+	+	+	+	+	+	-	+	+
genetic algorithm	-	+	-	+	-	+	-	+	-	+	-	+
software evolution	-	+	-	-	-	+	-	-	-	+	-	-
deliberative evolution	+	+	+	+	+	+	+	+	+	+	+	+

Table 1. Overview of various types of evolutionary processes.

2.3 Mind-Matter Relationships

As explained in the previous section, in the context of evolutionary processes both mental and physical aspects may play a role. Often a cyclic relation exists between mental and physical aspects of evolutionary processes, as illustrated in Figure 1.

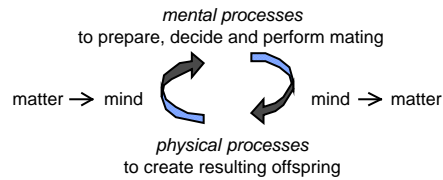


Figure 1. Cyclic relationship between mental and physical aspects of evolutionary processes.

Two transitions between mental (mind) to physical (matter) occur:

A transition from mind to matter.

Physical realisation of child individual(s) upon decision of parent individuals. This is a form of *mental causation*: mental states that have causal effects in the physical world; see, e.g., [18], Ch 6, pp. 125-154.

A transition from matter to mind.

The minds of child individuals emerge from their physical substrates. This can be related to the notion of *supervenience* in physicalism: mental properties supervene over physical properties in that for every mental property M that occurs, there exists some physical property P that also occurs, such that always if P occurs, also M occurs; e.g., [18], p. 223. In particular, if a parent has the possibility to be in a certain mental state M, then if he or she is in state M, a physical state P exists such that the parent is in state P, but also every other organism or physically embodied agent in state P is in state M. In particular, this holds for his or her children. If they are in the physical sense sufficiently equal to their parents, they will also have the possibility to be in physical state P, and therefore have mental state M. This shows how supervenience entails some form of inheritance of mentality over physical reproduction processes.

These considerations have their origin in the literature on Biology, Cognitive Science and Philosophy of Mind. The challenge of the work reported in this paper has been to exploit these conceptual analyses on deliberation, evolution and mind-matter relationships in order to come to a working software environment. Subsequent sections show how this has been accomplished.

3. The Global View

Two models play an important role in the design of an agent capable of deliberately creating or modifying agents

A. *A model for deliberation* about agent creation or modification, covering:

1 *Deliberation about the (intended) goal.*

The goal concerns the desired properties of an agent to be created or modified. As part of the deliberation process, generation of options, evaluation of options and decisions about options for these properties takes place.

2 *Deliberation about the intended creation or modification action.*

As part of this deliberation process, generation of options, evaluation of options and decisions about options for the structure of an agent to be created or modified satisfying the desired properties from A1 takes place.

3 *Deliberation about performing the intended creation or modification action.*

This deliberation process is about decisions to actually create or modify an agent according to the intended action (an agent structure) that was determined.

B. *A model for agent creation or modification* in the material world.

1 Execution of a creation or modification action that is performed by the agent.

2 Updating the world state to incorporate the intended effects of an executed creation or modification action.

The model for *deliberation* (A) is designed in such a way that the agent has:

- an explicit formal representation at a conceptual level of desired behavioural properties (requirements) of agents to be created or modified (supporting A1).
- an explicit formal representation at a conceptual level of (design) structures of an agent to be created or modified (supporting A2).
- knowledge to derive refinements of the requirements that are sufficiently specific to be related to specific design structures (supporting A1).
- knowledge to relate specific requirements to specific design structures of an agent to be created or modified (supporting A2).
- knowledge to make decisions on when to perform creation or modification actions (supporting A3).

The model for *modification* (B) is designed in such a way that:

- an agent executes intended creation or modification actions in the material world (supporting B1).
- the external world has a mechanism such that upon performance of a creation or modification action, it updates the world state on the basis of the structural information conveyed in the action, thus actually creating or modifying (while the multi-agent system is running) the new or existing agent and its programme code (supporting B2).
- the new or modified agents fully participate within the running multi-agent system after being created or modified by the material manipulations (supporting B2).

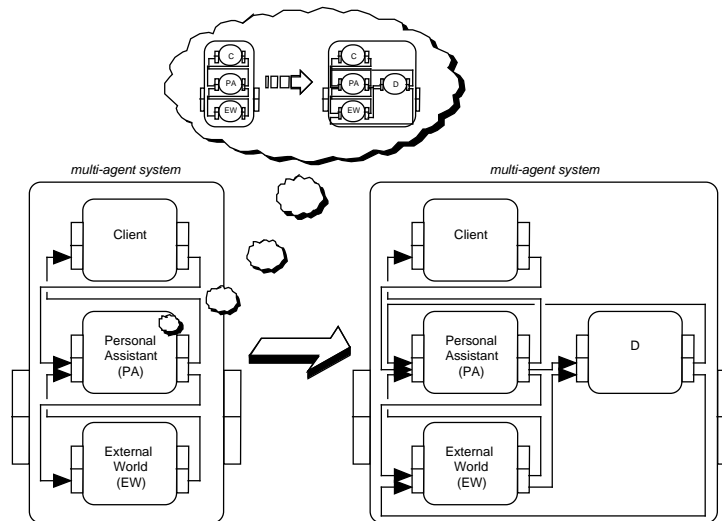


Figure 2. PA deliberately creates an agent. First PA obtains the material description of the multi-agent system. This description is re-designed by PA into a description, which includes a new agent (with the required properties). This description of the new multi-agent system is then realised, after which agent D participates in the multi-agent system.

In Figure 2 a sketch of the creation or modification process is depicted. The left box contains the multi-agent system before modification (consisting of the agents Personal Assistant, Client and External World), the right box after modification (with an additional agent D included). The Personal Assistant (PA) plays the role of the creating agent. It has internal (mental) representations of the multi-agent system before modification and designs a modification of the system by adding an agent D to the system. After this design process it executes the modification action in the External World, which represents all material aspects, including the material aspects of the agents.

4. A Model for Deliberation about Agent Creation

An illustrative pattern of deliberation within an agent is the following:

- an agent generates an intended goal to have a new or modified agent with particular desired or required properties (for example, by adopting this goal from another agent)
- on the basis of the intended goal, the agent determines refined properties for the new or modified agent, and generates more specific intended goals referring to the specific refined properties
- on the basis of the specific intended goals the agent determines a structure such that an agent with this structure would satisfy the specific properties to which the specific intended goals refer
- on the basis of this structure the agent generates an intended action: the action to create an agent with the structure as determined

This reasoning pattern is modelled in more detail by the *Generic Design Model* (GDM) introduced in [19]. A design model is applicable in this situation as intended goals and their related properties are viewed as (qualified) requirements and the structure (e.g., of an agent) is the description of the object of design.

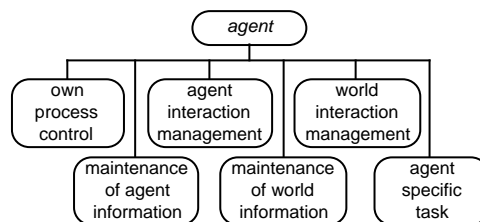


Figure 3. Generic Agent Model: the first level of the process composition.

The agent architecture proposed in this paper has been designed is a refinement of an existing Generic Agent Model GAM [20], depicted in Figure 3. This agent model supports the notion of a weak agent [3]: (1) it reasons about its own processes (component Own Process Control), supporting autonomy and pro-activeness, (2) it communicates (component Agent Interaction Management) with and maintains information (component Maintenance of Agent Information) about other agents, supporting social abilities, and reactiveness and pro-activeness with respect to other agents, and (3) it interacts (component World Interaction Management) with and maintains information (component Maintenance of World Information) about the external world, supporting reactiveness and pro-activeness with respect to the external world. The model is refined by specialisation of its component Agent Specific Task using a generic model for design. Moreover, the resulting model has been instantiated with ontologies and knowledge on the domain of agents.

4.1 The Deliberation Process as a Design Process

The compositional *Generic Design Model* (GDM) (see Figure 4) is a model of a complex reasoning process in which (1) explicit reasoning about requirements and their qualifications, (2) reasoning about design object descriptions, and (3) reasoning about the design process, are distinguished. This model is based on a logical analysis of design processes [21] and on analyses of applications, including elevator configuration [22] and design of environmental measures [23]. The model provides a generic structure, which can be refined for specific design tasks in different domains of application. Refinement of the generic design model, by specialisation and instantiation, involves the specification of more specific sub-processes, knowledge about applicable requirements and their qualifications, about the design object domain, and about design strategies.

An initial design problem statement is expressed as a set of initial requirements and requirement qualifications. *Requirements* impose conditions and restrictions on the structure, functionality and behaviour of the *design object* for which a structural description is to be generated during design. *Qualifications* of requirements are qualitative expressions of the extent to which (individual or groups of) requirements are considered hard or preferred, either in isolation or in relation to other (individual or groups of) requirements. At any one point in time during design, the design process focuses on a specific subset of the set of requirements. This subset of requirements plays a central role; the design process is (temporarily) committed to this current requirement qualification set: the aim of generating a design object description is to satisfy these requirements.

During design the considered subsets of the set of requirements may change, as may the requirements themselves. The same holds for design object descriptions representing the structure of the object to be designed. Figure 4 shows two levels of composition of the generic model for design. Three processes are

shown at the top level. One of these three processes, Design Object Description Manipulation, is shown to consist of four sub-processes.

The four processes (not decomposed in Figure 4) related to the process *requirement qualification set manipulation* are:

- RQS modification: the current requirement qualification set (RQS) is analysed, proposals for modification are generated, compared and the most promising (according to some measure) selected,
- deductive RQS refinement: the current requirement qualification set is deductively refined by means of the theory of requirement qualification sets,
- current RQS maintenance: the current requirement qualification set is stored and maintained,
- RQSM history maintenance: the history of requirement qualification sets modification is stored and maintained.

The four processes (see Figure 4) related to the process of *manipulation of design object descriptions* (DODM) are:

- DOD modification: the current design object description (DOD) is analysed in relation to the current requirement set, proposals for modification are generated, compared and the most promising (according to some measure) selected,
- deductive DOD refinement: the current design object description is deductively refined by means of the theory of design object descriptions,
- current DOD maintenance: the current design object description is stored and maintained,
- DODM history maintenance: the history of design object descriptions modification is stored and maintained.

The next three levels of the generic design model are described in [24].

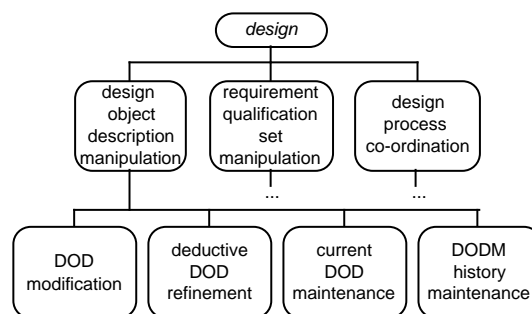


Figure 4. Generic Design Model: the sub-processes within the design and design object description manipulation processes

The reasoning model for design satisfies the criteria for deliberation put forward in Section 2. Manipulating sets of qualified requirements is, in this context, deliberation on goals, which includes generation of different options for possible goals, evaluation of each of these generated options for goals, and selecting one of the options for goals on the basis of these evaluations. Likewise, manipulation of design object descriptions is, in this context, deliberation on actions, which includes generation of different options for actions, evaluations of each of these options for actions (by using theories for deductive DOD refinement), and choosing one of the options for actions on the basis of these evaluations. Explicit co-ordination of the design process includes explicit strategic knowledge when to abandon a line of reasoning (e.g., possible actions for a temporary commitment to possible goals), and start another line of reasoning.

4.2 Ontologies and Knowledge involved in the Deliberation Process

The generic design model used in the agent can, in principle, be used for any domain of application. In this paper it is applied to the domain of compositional agent design. A formal ontology of properties of agents related to their structure or their behaviour has been developed: Section 4.2.1 discusses these properties.

Behavioural properties refer to the mental capabilities of an agent. Knowledge has been identified that can be used to reason about behaviour properties, to derive more specific behaviour properties by refining the original properties. These more specific properties play a crucial role in the design process: they guide the direction in which solutions are sought.

Moreover, a specific formal ontology for the structure of agents has been defined and is described below in Section 4.2.2. In this context, the structure of an agent is part of the 'physical' manifestation of an agent. The deliberation process employs this knowledge and these ontologies to arrive at a specific creation or modification action (i.e., with a specific structure specified) on the basis of the intended goal.

4.2.1 Representation of Properties of Agents

Requirements needed within the design process, are formulated in terms of abilities and properties of agents and the external world. Abilities and properties can be assigned to

- individual agents,
- the external world,
- an individual agent in relation to the agents and the world with which it interacts,
- the world in relation to the agents with which it interacts, and
- a multi-agent system as a whole.

Abilities of agents such as co-operation, bi-directional communication, and world interaction are often needed for agents to jointly be able to perform a certain task. In Figure 5 the ability of bi-directional communication and its refinements are depicted. For a description of other agent abilities see [25].

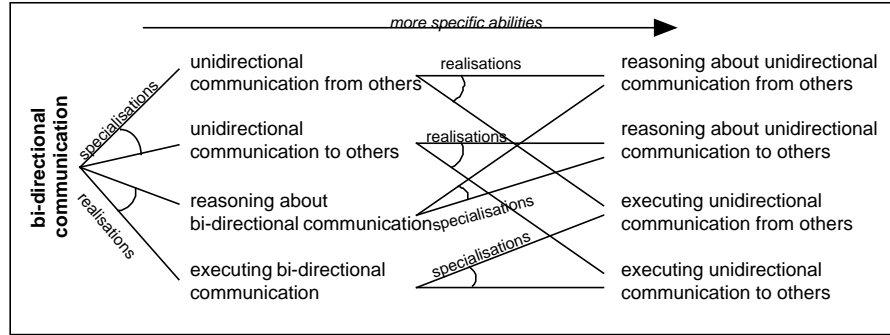


Figure 5. Refinements of the ability of bi-directional communication.

The ability of bi-directional communication can be refined, both with respect to its *specialisation* (refinement of the ability into more specific abilities) and with respect to its *realisation* (refinement of the ability into more fine-grained abilities related to reasoning about the ability, and more fine-grained abilities related to the effectuation of the ability).

Figure 5 shows the refinement relationships for the ability of bi-directional communication. The more specific abilities related to bi-directional communication are the ability to communicate to others (unidirectional communication to others) and the ability to receive communication from others (unidirectional communication from others). The abilities related to the realisation of the ability of bi-directional communication are the ability to *reason* about bi-directional communication, and the ability to *execute* bi-directional communication. These more specific abilities are further refined, and related to the ability to reason about unidirectional communication from others, the ability to reason about unidirectional communication to others, the ability to execute unidirectional communication from others, and the ability to execute unidirectional communication to others. Knowledge on refinements of the ability of bi-directional communication can be formally represented as in table 2.

Example Representation of requirements refinement knowledge

The knowledge element shown is the formal representation of refinement by specialisation of the ability ‘bi-directional communication’, shown in Figure 5.

```

if      is_qualified_requirement_selected_as_focus( QR: qualified_requirement_name )
and    holds( is_qualified_requirement(QR: qualified_requirement_name,
                                         Q: requirement_qualification,
                                         R: requirement_name )

and    holds( refers_to_requirement(      R: requirement_name,
                                         has_property( A: agent_name,
                                         is_capable_of_bidirectional_communication_
                                         with( A2: agent_name )) ),
                                         pos )

and    refinement_alternative( specialisations )
then    addition_to_current_RQS(
           is_qualified_requirement(      new_name( QR: qualified_requirement_name, a ),
                                         Q: requirement_qualification,
                                         new_name( R: requirement_name, a ) ) )

and addition_to_current_RQS(
           refers_to_requirement( new_name( R: requirement_name, a ),
                                   has_property( A: agent_name,
                                   is_capable_of_unidirectional_communication_
                                   from( A2: agent_name ) ) ) )

and addition_to_current_RQS(
           is_qualified_requirement(      new_name( QR : qualified_requirement_name, b ),
                                         Q: requirement_qualification,
                                         new_name( R: requirement_name, b ) ) )

and addition_to_current_RQS(
           refers_to_requirement( new_name( R: requirement_name, b ),
                                   has_property( A: agent_name,
                                   is_capable_of_unidirectional_communication_
                                   to( A2: agent_name ) ) ) )

and addition_to_current_RQS(
           is_qualified_requirement(      new_name( QR: qualified_requirement_name, c ),
                                         Q: requirement_qualification,
                                         new_name( R: requirement_name, c ) ) )

and addition_to_current_RQS(
           refers_to_requirement( new_name( R: requirement_name, c ),
                                   has_property( A: agent_name,
                                   is_capable_of_combining_unidirectional_
                                   communication_from_and_to( A2: agent_name ) ) );

```

Table 2. Example of Representation of requirements refinement knowledge.

Deliberation is employed to decide which refinement alternative should be employed for which ability. The possible refinements are the alternative options. This knowledge is located within the process RQS modification, see Figure 4.

Top-level requirements are refined into more specific requirements during a design process. The result is the construction of a specific hierarchy of requirements, which adheres to the requirements ontology and refinement knowledge. Figure 6 shows an example of (part of) such a requirements refinement hierarchy. The current prototype design agent makes extensive use of the requirements ontology, generic models and design object building blocks. Within the deliberation process only a limited number of options are generated and selected.

The most refined requirements are almost directly operationalisable by building blocks for design object descriptions. E.g., a specific design requirement, currently in focus in DOD modification, is broken up (i.e., refined) into smaller properties: assessment points. These assessment points can be tested for, and when not yet realised, building blocks related to an assessment point can be added to the current design

object description. The deliberation process explicitly employs strategic knowledge to generate options, this is modelled similar to modelling explicit strategies in the design model [26].

```

|_has_property( agent_D, is_capable_of_active_observation_in( world_W ) )
|_has_property( agent_D, is_capable_of_processing_observation_results_from( world_W ) )
|_has_property( agent_D, is_capable_of_reasoning_about_processing_observation_results_from( world_W ) )
|_has_property( agent_D, is_capable_of_executing_processing_observation_results_from( world_W ) )
|_has_property( agent_D, is_capable_of_combining_reasoning_about_and_executing_processing_observation_results( world_W ) )
|_has_property( agent_D, observation_initiation_in( world_W ) )
|_has_property( agent_D, is_capable_of_reasoning_about_observation_initiation_in( world_W ) )
|_has_property( agent_D, is_capable_of_executing_observation_initiation_in( world_W ) )
|_has_property( agent_D, is_capable_of_combining_reasoning_about_and_executing_observation_initiation_in( world_W ) )
|_has_property( agent_D, is_capable_of_combining_processing_observation_results_and_observation_initiation_in( world_W ) )

```

Figure 6. Part of a requirement refinement hierarchy constructed by the design agent.

4.2.2 Representation of the Structure of Agents

The implication of designing (parts of) a multi-agent system, is that a multi-agent system is the object of design, and as such its structure should be formally represented in a design object description. In this paper the design object structure is assumed to be a compositional. The assumption underlying this decision is that a compositional structure facilitates the process of (re-)design. The compositional formal specification language underlying DESIRE forms an adequate basis for such a design object description representation.

The description of the compositional structure is augmented with a description relating existing structures to generic models, as illustrated in table 3. This provides information useful for documentation purposes and it also provides valuable information for the identification of abilities and properties.

Example Representation of an agent design

The agent needs a representation of a multi-agent system including agents and the external world. Part of the top level of the multi-agent system can be represented as follows:

```
is_top_level(c_00)

corresponds_with(c_00, mas_S)

corresponds_with(c_01, agent_A)

corresponds_with(c_04, world_W)

has_characterisation(c_00, generic, multi_agent_system)

has_characterisation(c_01, generic, agent)

has_characterisation(c_4, generic, external_world)

corresponds_with(lm_01, active_observations)

has_subcomponent(c_00, c_01)

subcomponent(c_00, c_04)

has_information_link(c_00, lm_01)

has_source_component(lm_01, c_01)

has_destination_component(lm_01, c_04)
```

Unique identifiers (e.g., 'c_01') are assigned to components and information links so that names of links and components (e.g., 'agent_A') can be reused in several parts of the composition.

Table 3. Example of representation of an agent design

5. A Model for Agent Creation

After the deliberation on the creation process has been completed, and an agent decides to actually perform the modifications to an existing agent, or create a new agent, a number of steps take place. This final stage of the creation process can be compared to the realisation (or implementation) of the design object description into a 'real' artefact.

Creation/modification of an agent consists of two transitions between the mental and physical aspects. In the first transition, from mind to matter, a new agent's material representation is created (Section 5.1). In the second transition, from matter to mind, the new agent 'comes alive' in the world and may interact with other agents (Section 5.2).

5.1 Preparation of the Modification/Creation

Comparable to the approach put forward in [27], accomplishing the modification can be modelled by changing the material representation of a multi-agent system within the external world, as illustrated in table 4.

<p>Example Changing the material representation of a multi-agent system.</p> <p>The resulting design object description, <code>dod_55</code>, contains the complete set of modifications that are to be made to the multi-agent system <code>mas_S</code> (including the creation of a new agent). The design object properties that together form <code>dod_55</code>, are represented by statements such as:</p> <pre>has_DOD_characteristic(dod_55, corresponds_with(c_05, agent_D))</pre> <pre>has_DOD_characteristic(dod_55, has_subcomponent(c_00, c_05))</pre> <p>These statements reside at a meta-level with respect to design object description statements. The second argument of each statement expresses relationships within the design object description.</p>
--

Table 4. Example of changing the material representation of a multi-agent system

The modification action, and the structural changes in the material representation of the multi-agent system to which it refers changes are transferred from the agent to the external world. An example is given in table 5.

<p>Example Modification action</p> <p>Within the agent an intended action is generated to achieve the appropriate modification of the current multi-agent system:</p> <pre>to_be_performed(modify_according_to(dod_55));</pre>
--

Table 5. Example of modification action.

5.2 Execution of the Modification/Creation Action within the External World

To execute the creation action in the external world, the external world needs to have certain properties. These properties are related to how “equipped” the world is to handle interaction with agents. There are two generic properties needed for the interaction of agents with the external world: the processing of explicit observations and the processing of actions. Observation of the external world was needed to inform agent A of the current material representation of the multi-agent system.

The property of processing actions can be refined into the properties:

- the external world can receive initiated actions, and the related information, and
- the external world can perform actions (achieving the physical effects of actions).

To change the number of agents and their characteristics, the external world has to adapt the executable specification of that system while the multi-agent system is running. This implies that the parts of the system that are affected by the modifications are interrupted, their information states stored, after which the executable specification of those parts are modified, and the modified system reactivated with the correct information states. This is illustrated for the example in table 6 and was indeed achieved in the running prototype.

Example Result of the modification action.

The external world world_W executes the modification action and modifies the multi-agent system according to the given modifications.

As an agent in the multi-agent system, agent D receives a request from agent A: would it like to find out more about YYY?. The agent D gathers information on subject YYY by initiating observations in the world W, and interpreting the observation results. Once the answer is found, agent D reports its findings to agent A. Agent A can then finally answer the question of the client.

Table 6. Example of results of a modification action.

6. Discussion and Future Research

Results presented in this paper are discussed in Section 6.1, after which directions for further research are described in Section 6.2.

6.1 Discussion

Research within multi-agent systems research has focussed on the behaviour of individual agents and their interaction. The dynamic creation of new agents within an existing multi-agent system, on the basis of the identification of newly required functionality and behaviour, is an area on which little research has focussed. Most of the research in the area of dynamic agent creation is based on a *genetic programming approach*; e.g., [1,2]. These approaches do not take into account the possibility that agents deliberately influence their own evolution.

The approach taken in this paper is that an existing agent is capable of deliberately designing a new agent tuned to the needs as perceived and then be capable of bringing this agent to life. As genetic programming approaches in principle are based on manipulation of the material representations of the agents within the material world (outside the mental world of the agents), the approach introduced here

exploits the duality between the potential of deliberation on the one hand, and material manipulation possibilities on the other hand. In this sense the approach introduced adds to the genetic programming approach the possibility of a mental perspective from (individual or societies of) agents.

Recent approaches (e.g., [28,29]) have addressed the evolutionary synthesis of intelligent agents by a combination of neural networks, genetic algorithms and world simulation. These methods go beyond standard genetic algorithms in that the offspring has the form of agents equipped with a neural network that after their birth functioning in a certain task that requires some intelligence. Selection is done on task performance after some allowed training period. This approach certainly has potential. The profile of the approach measured in the criteria used in Section 2.2 is that selection has an endogenous aspect (but the threshold of acceptable functioning is put in an exogenous manner: a global fitness function). Generation of alternatives is exogenous. In common with our approach is that some mental capabilities are inherited in an embodied form (in their case by the neural network structures, in our case by design structures including reasoning models). A main difference is that in their case mental aspects play no direct role in generation of alternatives and in determination of selection criteria.

To design an agent capable of deliberatively creating new agents, insight is required in the type of agent to be designed and the deliberation model to be used. The architecture of the agent introduced here is based on an existing generic agent model, and includes a refinement of a generic model of design. It combines results from the area of Multi-Agent Systems and the area of AI and Design; e.g., [30,31]. In this paper a compositional approach to agent design has been followed. An agent's abilities are related to the tasks an agent is able to perform. In addition, the properties of the multi-agent system and the external world are of importance. As such, this work is related to the properties distinguished with respect to problem solving methods [32,33,34]. Within the field of Knowledge Engineering properties of problem solving methods are used to support knowledge engineers during the design process: providing a means to describe existing generic components that may be used, modified or refined during a design process, depending on their applicability in a given situation.

The Generic Agent Model (GAM) has been developed on the basis of experience with the design of agent models of different kinds; for example, models for information gathering agents, co-operative agents for project co-ordination, BDI-agents (deliberative agents with Beliefs, Desires, and Intentions), negotiating agents, broker agents, and agents simulating animal behaviour. The Generic Design Model (GDM) was developed and evaluated on the basis of experience with design applications in a number of domains; for example, design of sets of measures for environmental policy, aircraft design, and elevator design.

The refined model includes formalisations of the (compositional) structure of agents and (desired or required) properties of agents, and formalisations of agent design knowledge. A prototype design agent

has been implemented, which has demonstrated the re-design of its multi-agent systems and the realisation of the results: automated creation of agents.

A principled approach to the design of the architecture is of crucial importance: a generic conceptual architecture of an agent is needed to support the (re-)design process needed for dynamic creation or modification of agents based on dynamically imposed requirements. An approach in which conceptual design is the basis for structure-preserving (formal) detailed design and operational design, has turned out to be appropriate, as shown by the prototype implementation.

6.2 Future Research

Electronic Commerce necessarily involves interaction between human users in different types of organisations, and very dynamic automated environments, in which the parties involved are not known beforehand, and often change. In such environments human users can be supported by Personal Assistant Agents, which in turn make use of existing broker agents and other task-specific agents. Co-operation between these (human and computer) agents is to the advantage of all. To cope with the dynamic character of the environment, frequently new agents need to be created, or existing agents need to be modified for specific purposes. Such frequent modification of an environment necessitates almost continuous maintenance.

On the basis of the approach introduced in this paper, a generic multi-agent Electronic Commerce environment is being developed in which a broker agent can dynamically reconfigure (parts of) the multi-agent system by adding or modifying Personal Assistant agents, broker agents and additional agents. More specifically, the aim is to develop a multi-agent broker architecture with a number of co-operative broker agents, Personal Assistant agents, and task specific agents. Each broker agent can dynamically configure and implement new agents or modify existing agents as part of the multi-agent system as follows:

- if new users (clients) subscribe to a broker agent, Personal Assistant agents tuned to the requirements imposed by this user, may be created, or existing Personal Assistant agents may be modified, due to changed requirements.
- if required in view of the load of an existing broker agent, new broker agents can be added to distribute the load (and avoid overload of the existing broker agent), or existing broker agents can be modified.
- if opportune, or requested, new agents may be created to perform specific tasks, fulfilling certain dynamically imposed requirements; for example, for searching the Internet for specific types of information, or shadowing information at a specific site.

A few applications of broker agents have been built for this purpose; see, for example [35, 36, 37, 38, 39, 40]. However, these applications have been implemented without an explicit design at a conceptual

level, and without taking into account the dynamic requirements imposed by the domain of application and the maintenance problem implied by this dynamic character.

Acknowledgements

The authors wish to thank Pieter van Langen for his contributions to the Generic Design Model (GDM), and Lourens van der Meij and Frank Cornelissen for their support of the DESIRE software environment. This research has been supported by NWO within project 612-322-316: 'Evolutionary design in knowledge-based systems' (REVISE).

References

1. K. Cetnarowicz, M. Kisiel-Dorohinicki and E. Nawarecki, The Application of Evolution Process in Multi-Agent World to the Prediction System (M. Tokoro, Ed), Proceedings of the Second International Conference on Multi-Agent Systems, ICMAS'96, AAAI press, Menlo Park CA, 1996, 26-32.
2. C. Numaoka, Bacterial Evolution Algorithm for Rapid Adaptation (W. Van de Velde and J. W. Perram, Eds), Proceedings of the 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'96, Lecture Notes in Artificial Intelligence 1038, Springer Verlag, 1996, 139-148.
3. M. J. Wooldridge and N. R. Jennings, Intelligent Agents: Theory and Practice, Knowledge Engineering Review 10(2), 1995, 115-152.
4. A. S. Rao and M. P. Georgeff, Modelling Rational Agents within a BDI-Architecture (J. Allen, R. Fikes and E. Sandewall, Eds), Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, (KR'91), Morgan Kaufmann, 1991, 473-484.
5. F. M. T. Brazier, B. M. Dunin-Keplicz, J. Treur and L. C. Verbrugge, Modelling Internal Dynamic Behaviour of BDI agents (J. J. Meijer and P. Y. Schobbens, Eds), Formal Models of Agents: ESPRIT project MogelAge final workshop, Lecture Notes in AI 1760, Springer Verlag, 1999, 36-56.
6. F. M. T. Brazier, B. M. Dunin-Keplicz, N. R. Jennings and J. Treur, Formal Specification of Multi-Agent Systems: a Real World Case (V. Lesser, Ed), Proceedings First International Conference on Multi-Agent Systems, ICMAS'95, MIT Press, 1997, pp. 25-32; extended version (M. Huhns and M. Singh, Eds), International Journal of Co-operative Information Systems, IJCIS, , special issue on Formal Methods in Co-operative Information Systems: Multi-Agent Systems 6(1), 1997, 67-94.
7. S. Freeman and J. C. Herron, Evolutionary Analysis, Prentice Hall, 2001.
8. M. W. Strickberger, Evolution, 3rd Ed. Jones & Bartlett Publ, 2000.
9. C. Darwin, On The Origin of Species by Means of Natural Selection, or The Preservation of Favoured Races in the Struggle for Life, 1859.

10. L. Lefebvre, Feeding Innovations and their Cultural Transmission in Bird Populations (C. Heyes and L. Huber, Eds), *The Evolution of Cognition*, MIT Press, 2000, 311-328.
11. P. J. Richardson and R. Boyd, Climate, Culture, and the Evolution of Cognition Populations (C. Heyes and L. Huber, Eds), *The Evolution of Cognition*, MIT Press, 2000, 329-346.
12. B. G. Jr. Galef, Tradition in Animals: Field Observations and Laboratory Analyses (M. Bekoff and D. Jamieson, Eds), *Readings in Animal Cognition*, MIT Press, 1996, 91-106.
13. M. D. Hauser, *The Evolution of Communication*, MIT Press, 1997.
14. M. Philips and S. N. Austad, Animal Communication and Social Evolution Analyses (M. Bekoff and D. Jamieson, Eds), *Readings in Animal Cognition*, MIT Press, 1996, 257-268.
15. Th. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*, Oxford University Press, 1996.
16. M. M. Lehman and L. A. Belady (Eds), *Program Evolution*, Academic Press, APIC Studies in Data Processing No 27, 1985.
17. J. C. van Vliet, *Software Engineering: Principles and Practice*, John Wiley & Sons, Chicester, 1993.
18. J. Kim, *Philosophy of Mind*, Westview Press, 1996.
19. F. M. T. Brazier, P. H. G. van Langen, Zs. Ruttkay and J. Treur, On formal specification of design tasks (J. S. Gero and F. Sudweeks, Eds), *Artificial Intelligence in Design '94*, Kluwer Academic Publishers, Dordrecht, 1994, 535-552.
20. F. M. T. Brazier, C. M. Jonker and J. Treur, Compositional Design and Reuse of a Generic Agent Model, *Applied Artificial Intelligence Journal* 14, 2000, 491-538.
21. F. M. T. Brazier, P. H. G. van Langen and J. Treur, A logical theory of design (J. S. Gero, Ed), *Advances in Formal Design Methods for CAD*, Chapman & Hall, New York, 1996, 243-266.
22. F. M. T. Brazier, P. H. G. van Langen, J. Treur, N. J. E. Wijngaards and M. Willems, Modelling an elevator design task in DESIRE: the VT example (A. Th. Schreiber and W. P. Birmingham, Eds), *Special Issue on Sisyphus-VT*, *International Journal of Human-Computer Studies* 44, 1996, 469-520.
23. F. M. T. Brazier, J. Treur and N. J. E. Wijngaards, Interaction with experts: the role of a shared task model (W. Wahlster, Ed), *Proceedings European Conference on AI (ECAI '96)*, Wiley and Sons, Chichester, 1996, 241-245.
24. F. M. T. Brazier, C. M. Jonker, J. Treur and N. J. E. Wijngaards, Compositional Design of a Generic Design Agent, *Design Studies journal*, to appear, 2001.
25. F. M. T. Brazier, C. M. Jonker, J. Treur and N. J. E. Wijngaards, The Role of Abilities of Agents in Re-design (B. Gaines and M. Musen, Eds), *Proc. of the 11th Knowledge Acquisition Workshop, KAW'98*. Banff. University of Calgary, URL: <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/brazier2>, 1998.
26. F. M. T. Brazier, P. H. G. van Langen and J. Treur, Strategic Knowledge in Compositional Design Models (J. S. Gero and F. Sudweeks, Eds), *Proceedings of the Fifth International Conference on Artificial Intelligence in Design, AID'98*, Kluwer Academic Publishers, Dordrecht, 1998, 129-147.

27. C. M. Jonker and J. Treur, Modelling an Agent's Mind and Matter, Proceedings of the 8th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'97, Lecture Notes in AI 1237, Springer Verlag, 210-233.
28. K. Balakrishnan and V. Honavar, Evolutionary and Neural Synthesis of Intelligent Agents (M. Patel, V. Honavar and K. Balakrishnan, Eds), Evolutionary Synthesis of Intelligent Agents, MIT Press, Cambridge, MA, 2001, 1-27.
29. K. Balakrishnan and V. Honavar, Evolving Neuro-Controllers and Sensors for Artificial Agents (M. Patel, V. Honavar and K. Balakrishnan, Eds), Evolutionary Synthesis of Intelligent Agents, MIT Press, Cambridge, MA, 2001, 109-152.
30. R. D. Coyne, M. A. Rosenman, A. D. Radford, M. Balachandran and J. S. Gero, Knowledge-based design systems, Addison-Wesley Publishing Company, Reading, 1990.
31. T. Smithers, On knowledge level theories of design process (J. S. Gero and F. Sudweeks, Eds), Artificial Intelligence in Design '96, Kluwer Academic Publishers, Dordrecht, 1996, 561-579.
32. R. Benjamins, D. Fensel and R. Straatman, Assumptions of Problem-Solving Methods and their Role in Knowledge Engineering (W. Wahlster, Ed), Proceedings European Conference on AI (ECAI '96), Wiley and Sons, Chichester, 1996, 408-412.
33. J. A. Breuker, Problems in indexing problem solving methods (D. Fensel, Ed), Proceedings of the Problem Solving Methods for Knowledge Based Systems workshop, IJCAI'97, 1997, 19-35.
34. D. Fensel, E. Motta, S. Decker and Z. Zdrahal, Using Ontologies for Defining Tasks, Problem-Solving Methods and their Mappings (E. Plaza and R. Benjamins, Eds), Knowledge Acquisition, Modelling and Management, proceedings of the 10th European workshop, EKAW'97, Lecture Notes in Artificial Intelligence 1319, Springer, Berlin, 1997, 113-128.
35. A. Chavez and P. Maes, Kasbah: An Agent Marketplace for Buying and Selling goods, Proc. of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, PAAM'96, The Practical Application Company Ltd, Blackpool, 1996, 75-90.
36. A. Chavez, D. Dreilinger, R. Gutman and P. Maes, A Real-Life Experiment in Creating an Agent Market Place, Proc. of the Second International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, PAAM'97, The Practical Application Company Ltd, Blackpool, 1997, 159-178.
37. D. Kuokka and L. Harada, On Using KQML for Matchmaking (V. Lesser, Ed), Proc. of the First International Conference on Multi-Agent Systems, ICMAS'95, MIT Press, Cambridge, MA, 1995, 239-245.
38. D. Martin, D. Moran, H. Oohama and A. Cheyer, Information Brokering in an Agent Architecture, Proc. of the Second International Conference on the Practical Application of Intelligent Agents and Multi-agent Technology, PAAM'97, The Practical Application Company Ltd, Blackpool, 1997, 467-486.
39. T. Sandholm and V. Lesser, Issues in Automated Negotiation and Electronic Commerce: Extending the Contract Network (V. Lesser, Ed), Proc. of the First International Conference on Multi-Agent Systems, ICMAS'95, MIT Press, Cambridge, MA, 1995, 328-335.

40. M. Tsvetovatyy and M. Gini, Toward a Virtual Marketplace: Architectures and Strategies, Proc. of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, PAAM'96, The Practical Application Company Ltd, Blackpool, 1996, 597-613.